

ORIGINAL PAPER

A aplicação da derivada na detecção de bordas para alunos do primeiro período de engenharia

First Author¹, Second Author², Third Author² and Fourth Author²¹First Institution and ²Second Institution

*first@uni.edu; second@lab.edu; ...

Received: yyyy-mm-dd. Revised: yyyy-mm-dd. Accepted: yyyy-mm-dd.

Abstract

The Abstract (250 words maximum) should be structured to include the following details: **Background**, the context and purpose of the study; **Results**, the main findings; **Conclusions**, brief summary and potential implications. Please minimize the use of abbreviations and do not cite references in the abstract.

Key words: Keyword1; keyword 2; keyword 3 (Three to five keywords representing the main content of the article, in alphabetic order)

Resumo

O Resumo (com até 250 palavras) deverá ser estruturado para abordar os seguintes detalhes: **Background**, o contexto e propósito do estudo; **Resultados**, os principais encontrados; **Conclusões**, um breve resumo do trabalho e as implicações em potencial. Por favor, evite, na medida do possível, o uso de abreviaturas e não cite referências no resumo.

Palavras-Chave: Palavra-chave 1; palavra-chave 2; palavra-chave 3 (De três a cinco palavras-chave que representem os principais tópicos do artigo, em ordem alfabética)

1 Introduction to this Template

The Revista Brasileira de Computação Aplicada (RBCA) is a **open access journal** linked to the [Graduate Program in Applied Computing \(PPGCA\)](#) of the [University of Passo Fundo](#), Brazil. RBCA aims to provide to the scientific community article that present an interdisciplinary perspective of the application of Computing in different areas of knowledge.

This is the \LaTeX template for RBCA journal manuscript submissions. **Submissions that do not use the format available in this template will be automatically rejected.**

Articles submitted to RBCA should be between 8 and 15 pages and will be published electronically. The languages accepted by RBCA are Portuguese and English. We alert the authors that the preparation of the manuscript should be made carefully, both in its

scientific content and in its grammatical correctness.

There are important commands in the preamble that you will need to modify for your own manuscript. Specify your manuscript's category with the `\papercat{...}` command in the preamble. See the sample code in the preamble for a sample of how author, affiliation and e-mail information can be specified. Information about this edition of RBCA and publication details of the manuscript will be filled out by the journal's editor.

This template has been edited and validated in the [TexStudio](#)© program and the [Overleaf](#)© Collaborative Writing and Publishing System. Any questions or problems regarding this template should be reported to the e-mail of the journal (rbca@upf.br).

The remainder of this current section will provide some sample \LaTeX code for various elements you may want to include in your manuscript.

2 Processamento de imagem

O processamento de imagem atua na separação de pontos de interesse por meio de operações que resultaram em uma imagem de saída. Utilizando imagens é possível extrair informações sobre as características físicas e geométricas básicas dos objetos, tais como a dimensão, a área, o perímetro e a forma.

Segundo [Gonzalez and Woods \(2010\)](#) uma imagem pode ser representada como uma função bidimensional denotada por $f(x,y)$, tal que x e y são coordenadas de um plano e f expressa a intensidade de cinza nas coordenadas. Existe um número finito de elementos no contra domínio dessa função e cada um é denominado como *Picture element (pixel)*.

Para que um sistema computacional reconheça os elementos de uma imagem é necessário a digitalização de cada pixel. Para isso são necessários dois procedimentos, primeiramente a amostragem e então a quantização. A amostragem consiste na seleção de pontos igualmente espaçados de uma função contínua, que representa uma imagem, de forma que esses pontos sejam coletados gerando um conjunto de localizações discretas. Por outro lado, a quantização transforma os valores contínuos da intensidade de cores uma determinada amostra em valores discretos. Esses procedimentos possibilitam a alocação dos valores quantizados em uma matriz nas posições advindas do processo de amostragem.

O armazenamento em uma matriz tem como vantagem a possibilidade da aplicação de operações matriciais sobre a fotografia. Nesse caso, os pixels são representados por uma coordenada (x, y) referente à sua posição no plano, e z , a intensidade da cor. Na área de computação visual adotou-se que a intensidade do pixel igual a 0 representaria o preto e o 255 representaria o branco, todas as outras cores são variações nesse intervalo.

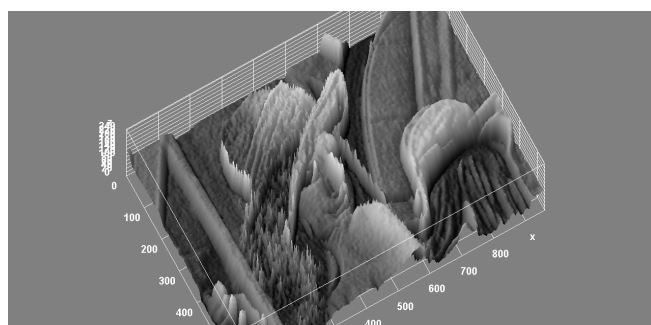


Figure 1: Representação das dimensões da imagem e a intensidade de cada pixel

As imagens são, normalmente, representadas no sistema RGB (do inglês Red, Green and Blue), isso significa que cada pixel é uma aproximação resultante das intensidades de vermelho, azul e verde. A união das três cores em cada pixel forma uma imagem de 3 bandas ([Biasi et al.; 2002](#)). Segundo [de Queiroz and Gomes \(2006\)](#) a demonstração da imagem de três bandas pode ser a seguinte equação, tal que F_r , F_g e F_b representam respectivamente a intensidade das bandas vermelha, verde e azul:

$$F(x, y) = F_r(x, y) + F_g(x, y) + F_b(x, y) \quad (1)$$

A par desses conceitos, o processamento se inicia transformando a imagem formada pelas bandas RGB em tons de cinza. Isso porque as variações entre as bandas serão analisadas apenas como uma variação de cinza. Segundo a documentação oficial dos desenvolvedores da biblioteca OpenCV, incluída no algoritmo, a transformação aplicada é baseada em uma média ponderada das componentes RGB de cada pixel, dada pela fórmula a seguir:

$$F(x, y) = (0,299 * F_r(x, y)) + (0,587 * F_g(x, y)) + (0,114 * F_b(x, y)) \quad (2)$$



Figure 2: Representação de uma imagem nas bandas RGB e em escala de cinza

Segundo [de Queiroz and Gomes \(2006\)](#) existem duas classificações importantes quando se trata de operações que manipulam de forma direta o pixel: as operações pontuais e as operações locais ou por máscaras. Nas operações pontuais cada pixel é tratado de forma independente. Já nas operações locais ou por máscara o valor de saída depende de um conjunto de pixels vizinhos. É importante salientar, que as máscaras são matrizes com dimensões pequenas em que, o pixel a ser tratado é posicionado no centro da matriz de operação e o resultado é um novo valor na mesma posição.

A par desses conceitos é possível, então, aplicar o método para a detecção de borda desenvolvido por John Canny. A imagem de saída resultante da aplicação desse método é do tipo imagem binária, pois os elementos variam entre 0 ou 1, de forma que 1 representa que naquele pixel há uma borda.



Figure 3: Representação da borda de uma imagem

3 Métodos Canny para detecção de bordas

3.1 Suavização Gaussian

3.2 Detecção do gradiente

Após a suavização de ruídos na imagem, os métodos de detecção das bordas precisam de ferramentas que proporcionem encontrar a localização de variações bruscas das cores. É nesse momento que os conceitos de derivadas são utilizados, por meio do gradiente.

O gradiente é um vetor bidimensional, denotado por ∇f , que representa a taxa de variação da função f na posição (x,y) (Gonzalez and Woods; 2010, p. 465). É necessário utilizar esse tipo de ferramenta uma vez que, como foi visto, a imagem pode ser vista como uma função de duas dimensões, podendo ocorrer variações de intensidade cores tanto em relação ao eixo x quanto ao eixo y .

$$\nabla f = (G_x, G_y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) \quad (3)$$

No contexto do processamento de imagem, o gradiente fornecerá a direção e o sentido da variação de intensidades de cores entre um pixel e seus vizinhos por meio do ângulo α (4). A direção do gradiente é importante uma vez que ele é ortogonal à borda, como é possível visualizar na imagem 4. Além disso, a magnitude, dado por $M(x,y)$ (5), indicará a intensidade dessa variação.

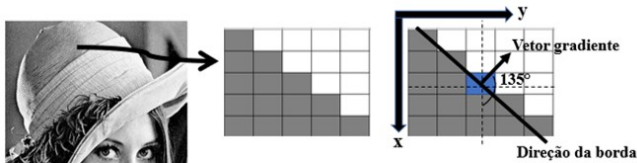


Figure 4: Vetor gradiente representado sobre a imagem

$$\alpha(x,y) = \arctan\left(\frac{G_x}{G_y}\right) \quad (4)$$

$$M(x,y) = \sqrt{(G_x)^2 + (G_y)^2} \quad (5)$$

O gradiente é uma importante ferramenta para encontrar as bordas, entretanto, como aplicar a derivada em um sistema com valores digitais? Para isso se utiliza uma aproximação da derivada parcial, que é aplicada pixel-a-pixel. Como já visto, a deriva é conceituada como a variação instantânea de uma dada função, generalizando esse conceito pode-se chegar as equações 6 e 7, representada matricialmente pela imagem(a) 5. Isso significa que a diferenciação em um elemento da imagem pode ser vista como a diferença das intensidades dos elementos vizinhos.

$$G_x = \frac{\partial f(x,y)}{\partial x} = f(x+1,y) - f(x,y) \quad (6)$$

$$G_y = \frac{\partial f(x,y)}{\partial y} = f(x,y+1) - f(x,y) \quad (7)$$

É importante salientar que essa forma de calcular o gradiente não é suficiente quando se trata de bordas diagonais, uma vez que a intensidade de cor dos pixels vizinhos na diagonal não são considerados. Uma forma de solucionar esse problema é construir uma máscara de 2 dimensões que detecte variações de todos os 8 vizinhos. Segundo Gonzalez and Woods (2010) essas máscaras recebe o nome de Operadores de Prewitt e pode ser vista na imagem(b) 5. O método de Canny utiliza o Operador de Sobel, o qual valoriza os pixel mais próximos por meio de um peso com valor 2, representado pela imagem(c) 5. Para exemplificar a aplicação do operador Sobel, o aplicaremos sobre a imagem(a) 6, uma figura constituída apenas por duas tonalidades de cinza, 0 e 255. Como só há duas cores, a variação só é detectada exatamente quando há a troca de cores, como pode ser visto na imagem(c) 6, que representa o gradiente na direção y , de tal forma que a variação é vista como uma linha branca traçada próximo do centro. Como não há variação de cor quando se analisa o eixo x a "deriva" é igual a zero, logo a imagem do gradiente em x (figura (b) 6) (G_x) é constituída de zeros, logo representada apenas por preto.

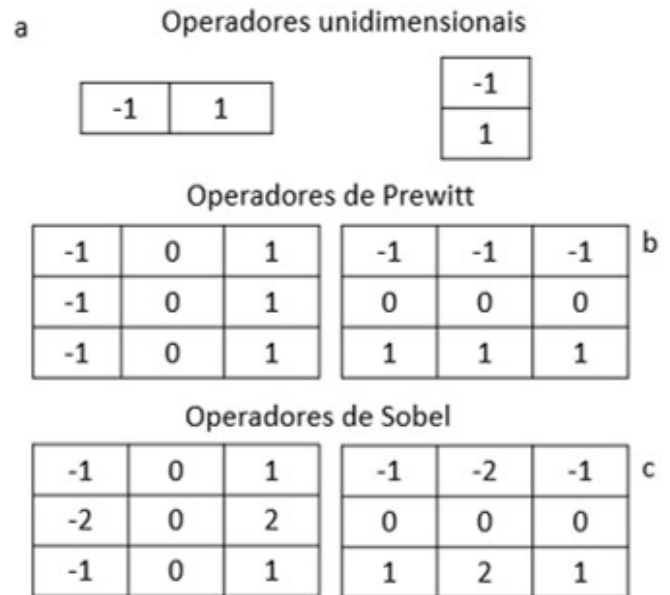


Figure 5: Matrizes representantes dos operadores diferenciais

Como forma de demonstrar o método de detecção da variações de cores, suponha que se deseja detectar a existência de uma borda no pixel destacado (na cor azul) da imagem 4. Suponha também que a cor escura seja igual a 0 e a cor clara seja 255. O primeiro

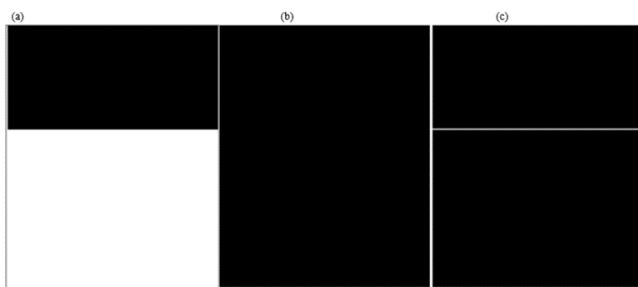


Figure 6: Representação da aplicação do gradiente na direção de x (b) e y (c)

passo é centralizar o pixel com o centro do operador, iremos utilizar o operador de Sobel, veja a imagem 7.

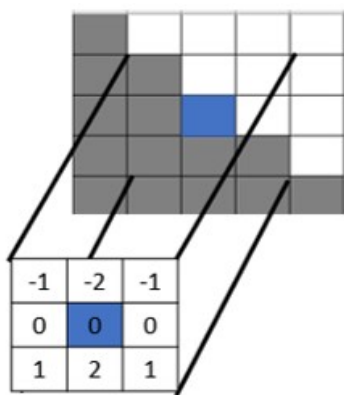


Figure 7: Centralizando pixel com o operador

Multiplique-se assim o pixel analisado e seus 8 vizinhos pelo valor correspondente à sua posição no operador. Esse processo resultará uma matriz com as mesmas dimensões do operador. O próximo passo é fazer o somatório dos elementos desse array resultante, detectamos dessa forma a intensidade do gradiente em uma das direções. Por fim, o valor obtido é colocado, na mesma posição do pixel que foi estudado, em uma nova matriz, denominada como matriz de gradientes. Observe esses processos na imagem 8.

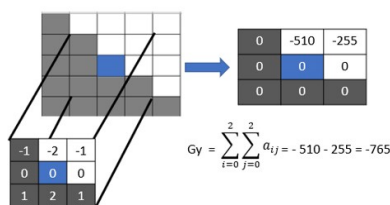


Figure 8: Encontrando a intensidade do gradiente

Realizando os mesmos procedimento com o operador em x, detectamos que $G_x = 765$. Utilizando as operações da equação 4, constatamos que $\alpha = -45^\circ$ o que equivale a 135° considerando o quadrante positivo. Por fim, usando a equação 5 é possível deduzir um valor representante da variação nas duas dimensões.

Realizando as contas obtemos que $M = 765 * \sqrt{2}$. Esse valor é alocado em uma nova matriz, na posição correspondente ao pixel estudado.

3.3 Supressão do não máximo

3.4 Limiarização

O processo final da detecção de bordas elaborado por Canny, a limiarização, consiste em separar a imagem em duas partes onde uma representa a região de interesse, a borda, e a outra parte representa o fundo da imagem. Esse processo se resume na identificação da diferença de cinza, através de um valor limiar, denominado binarização da imagem. [Morgan et. Al 2008]

Esse limiar traduz-se em limitar o máximo e mínimo das características de uma imagem, se tratando disso, os valores abaixo e acima desse limite são tratados de maneira diferentes, podendo ser interpretados como borda ou fundo. [Silva, 2000] Desse modo o processo de limiarização produz uma imagem bem definida, distinguindo o fundo e o objeto (área de interesse) através da diferença bem definida do nível de cinza da imagem, ou seja, a diferença de intensidade entre um pixel e outro é maior. Conforme Grando(2005) podemos definir a limiarização como: [Morgan et. Al 2008]

$$G(x,y) = \begin{cases} 1 & f(x,y) > T \\ 0 & f(x,y) \leq T \end{cases} \quad (8)$$

Onde $f(x,y)$ é a imagem de entrada, T é o valor de limiar e $g(x,y)$ é a imagem de saída, imagem limiarizada [Morgan et. Al 2008]. Podemos observar que na fórmula os valores maiores que o limiar são considerados como borda, e os valores abaixo desse limiar são considerados como fundo. Assim, a imagem produzida, $g(x,y)$, será uma imagem com o número de ruídos reduzidos.

4 Desenvolvimento do algoritmo

Como apresentado, o objetivo geral deste artigo é demonstrar de forma prática a aplicação do cálculo de derivadas na área da Engenharia de Computação. Com esse intuito, foram realizadas pesquisas e experimentações para aprofundar o conhecimento em processamento de imagens. Além disso, um dos objetivos específicos foi desenvolver um algoritmo capaz de encontrar as bordas de um objeto em uma imagem e com isso retornar dados da sua medição, demonstrando uma aplicação dos dados extraídos de uma imagem processada.

A detecção da variação de intensidade em imagens é algo canônico na visão computacional (Canny; 1983), e é baseado nos estudos do método aplicado por John Canny para detecção de bordas que foram realizadas as experimentações. Retomando o objetivo geral do trabalho, o entendimento do método Canny foi necessário, dado que este se baseia na aplicação da primeira derivada da função Gaussiana.

A primeira etapa foi dedicada a desenvolver um algoritmo que através da manipulação da matriz

de uma imagem, retornasse uma matriz resultante apenas com elementos binários. Utilizando a biblioteca OpenCV, foi desenvolvido um algoritmo que recebe uma imagem e a salva na forma matricial, onde cada pixel representa um elemento da matriz. O número atribuído a esse pixel, ou seja, ao elemento dessa matriz, é um valor do sistema RGB. Na sequência, transformam-se os valores dos canais RGB em um único representante na escala de cinza. Essa modificação é necessária para diminuir as variações entre as cores dos pixels. Por fim, é aplicado o método Canny, que é capaz de identificar mudanças na intensidade dos pixels da imagem para detectar a presença de bordas.

5 Resultados

Retomando um dos objetivos específicos deste trabalho, utilizamos a matriz binária gerada após aplicação do método Canny para realizar a medição de um objeto contido na imagem. Interpretando a matriz resultante como um sistema de coordenadas, onde as colunas representam o eixo x e as linhas representam o eixo y, foram criadas duas funções:

- A primeira função, busca pelo primeiro e pelo último elemento (x, y) da matriz cujo valor seja 1, ou seja, presença de borda. Subtraindo o componente y do último elemento, pelo componente y do primeiro elemento, obtêm-se a altura em pixels do objeto. Essa função limita-se a medir objetos sem que haja aplicação de rotações.

- A segunda função, faz uma varredura na matriz em sentidos opostos, encontrando duas quinas no objeto. Dessa maneira, com a informação da coordenada (x, y) das quinas, utilizou-se o conceito de distância entre dois pontos para encontrar o valor em pixels de uma quina à outra. Essa função limita-se a medir objetos com quinas bem definidas, quando a medida é o diâmetro, por exemplo, utiliza-se o método citado anteriormente. Rotações que não envolvam o eixo de profundidade são medidas sem que haja danos ao resultado.

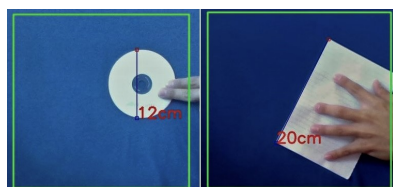


Figure 9: Demonstração dos métodos de medição

Foram realizadas diversas experimentações para constatar a correlação entre a medida em pixels, e a medida real do objeto em milímetros. Neste processo controlaram-se algumas variáveis, como: a distância do objeto à câmera; a rotação do objeto; e o processo de limiarização do método Canny.

Como pode ser observado na figura 10, o desvio padrão da relação pixel/milímetros é bem inferior àquela constatada na figura 11. Isso indica que quanto menor a distância entre a câmera e o objeto medido, maior será a repetitividade do processo, assim como melhor será a precisão. Isso está relacionado com a

Objeto	Tamanho (mm)	Medida em pixels	Relação Pixel/mm
A4 horizontal	210	280	0,75
A4 vertical	294	394	0,75
Celular	140	185	0,76
Cartão vertical	94	124	0,76
Cartão horizontal	66	88	0,75
Média:			0,75
Desv. Padrão:			0,005

Figure 10: Resultados das medições realizadas a 500mm de distância

Objeto	Tamanho (mm)	Medida em pixels	Relação Pixel/mm
A4 horizontal	210	142	1,48
A4 vertical	294	188	1,56
Celular	140	91	1,54
Cartão vertical	94	64	1,47
Cartão horizontal	66	45	1,47
Média:			1,49
Desv. Padrão:			0,045

Figure 11: Resultados das medições realizadas a 1000mm de distância

perda de detalhes do objeto ao afastá-lo da câmera.

Buscando realizar medições em tempo real, foi necessário criar uma etapa para estabelecer uma relação inicial entre pixels e centímetros a ser mantida durante a medição de novos objetos. Este processo consiste na utilização de um objeto com dimensões conhecidas, e a partir da medição deste objeto, dada em pixels, computar a relação entre pixel e centímetro a ser mantida.

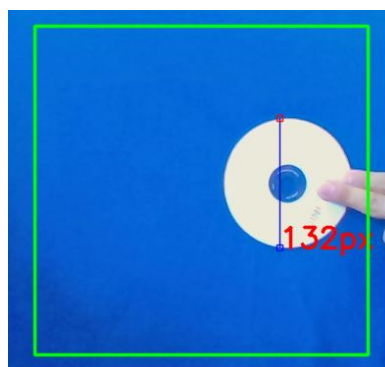


Figure 12: Computando a altura em pixels para ser utilizado como referência

A imagem 12, o diâmetro do CD foi equivalente à 132 pixels. Como um CD padrão tem diâmetro de 12,0 centímetros, o software computou a relação pixel/centímetro equivalente à aproximadamente 0,091 centímetros por pixel. Esta relação foi estabelecida com a câmera a uma distância de 750 milímetros do objeto, e, portanto, mantida para as demais medições.

Como as bordas que não representam o objeto, e as bordas do objeto são armazenadas da mesma maneira, ao analisar a imagem completa, a presença de detalhes indesejados impedia o processo de medição. Essa situação era devido ao fato de o algoritmo procurar o primeiro e o último elemento da matriz que representasse borda, e ao encontrar falsas bordas, ou bordas que não representam o objeto, falhas eram geradas. Percebido isso, foi desenvolvida uma nova função no programa com a função de delimitar a região de interesse. Esta função tem como objetivo

reduzir a área de busca por bordas, transformando a matriz da imagem em uma matriz menor.

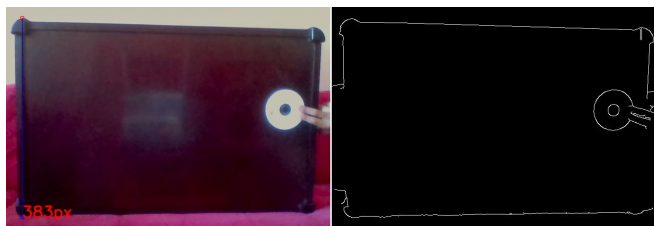


Figure 13: Detecção de borda sem a região de interesse

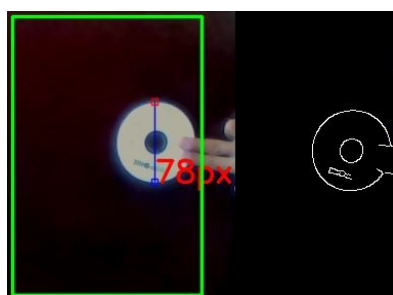


Figure 14: Detecção de borda dentro da região de interesse

Como pode ser observado na figura 13, em que não há delimitação da área de busca, o algoritmo retornou uma altura em pixels equivalente à 383px. Já na imagem 14, a região de busca por bordas corresponde à área interna do retângulo de contorno verde, isso significa que somente o que estiver compreendido na região de interesse será analisado.

Além do problema com a região de interesse, houve também a necessidade de permitir que o usuário pudesse modificar, quando necessário, a limiarização desejada para o processo. Pois, a aplicação da limiarização é difícil e envolve experimentação. “A permanência de falsas bordas, após a limiarização, pode ter como motivo a escolha de um limiar baixo, ou alto demais” (Do Vale and DAL POZ 2002,p. 9). Na imagem 14 as bordas ficam restritas ao objeto, já na figura 15, ocorre a presença de falsas bordas, sendo possível perceber o impacto de uma limiarização mal aplicada, o que interfere na medição.



Figure 15: Detecção de borda com limiarização mal aplicada

Por fim, reduzindo a região de interesse, e

permitindo a limiarização durante o processo de medição, o algoritmo foi capaz de retornar medições dos objetos com uma precisão de 0,0962 centímetro, ou seja, inferior à 1 milímetro (vide Figura 16). Esse é um resultado aceitável, visto que se buscava a demonstração de uma utilização dos dados de saída do método de detecção de borda. Porém, o fato de trabalharmos com uma câmera 2D, implicou na perda de percepção de profundidade. Logo, quando há rotação do objeto em profundidade a medição passa a se invalidar. Aprofundar os cálculos para encontrar medições do objeto independente da rotação, é uma das possíveis vertentes que este trabalho poderá seguir, portanto não será abordada essa variável de processo.

Objeto	Medida esperada (cm)	Medida encontrada (cm)	Erro (cm)
CD Padrão	12,00	12,09	0,09
Mini CD	8,00	8,03	0,03
Cartão de crédito Horizontal	5,40	5,50	0,10
Cartão de crédito Vertical	8,56	8,66	0,10
A4 Horizontal	21,00	21,11	0,11
A4 Vertical	29,70	29,77	0,07
Cartão de visita Horizontal	4,80	4,96	0,16
Cartão de visita Vertical	8,90	9,02	0,12
Média			0,0975

Tabela 4: Resultados das medições realizadas

Figure 16: Resultados de medições

References

- Biasi, H. H. D. et al. (2002). Desenvolvimento de uma metodologia de visão computacional para o auxílio no planejamento cirúrgico no implante de próteses endoluminais.
- Canny, J. F. (1983). A variational approach to edge detection., *AAAI*, Vol. 1983, pp. 54–58.
- de Queiroz, J. E. R. and Gomes, H. M. (2006). Introdução ao processamento digital de imagens., *RITA* 13(2): 11–42.
- Gonzalez, R. C. and Woods, R. C. (2010). *Processamento digital de imagens*, 3 edn, Gabriela Trevisan.